

What is claimed is:

1. A method for data retention in a data cache, comprising:
 - referencing dirty data stored in a cache in a first least recently used list;
 - and
 - referencing clean data in the cache in a second least recently used list;
 - wherein dirty data is destaged from the cache when it reaches the tail of the first least recently used list and clean data is purged from the cache when it reaches the tail of the second least recently used list.
2. A method as claimed in claim 1, wherein dirty data which is destaged to a data storage means and a copy of the data is retained in the cache as clean data is deleted from the first list and added to the second list.
3. A method as claimed in claim 1, wherein a read command which is a cache miss fetches data from a data storage means and the data is retained in the cache with a reference in the second list.
4. A method as claimed in claim 1, wherein the method includes keeping a flag with each data reference in the first list indicating whether or not the data has been read whilst on the first list.
5. A method as claimed in claim 1, wherein, if the data was read when referenced in the first list, the data is added to the head of the second list when the data is destaged.
6. A method as claimed in claim 1, wherein, if the data was not read when referenced in the first list, the data is either maintained in its position in the second list or discarded.

7. A method as claimed in claim 4, wherein the flag includes a timestamp each time the data is read and the timestamp is used to prioritise the position of the data reference in the second list.
8. A method as claimed in claim 1, wherein data is partly dirty and partly clean and is referenced in both the first and second lists.
9. A data storage system comprising:
 - a storage controller including a cache;
 - a data storage means; and

the cache has a first least recently used list for referencing dirty data which is stored in the cache, and a second least recently used list for referencing clean data;

wherein dirty data is destaged from the cache when it reaches the tail of the first least recently used list and clean data is purged from the cache when it reaches the tail of the second least recently used list.
10. A data storage system as claimed in claim 9, wherein dirty data which is destaged to a data storage means and a copy of the data is retained in the cache as clean data is deleted from the first list and added to the second list.
11. A data storage system as claimed in claim 9, wherein a read command which is a cache miss fetches data from the data storage means and the data is retained in the cache with a reference in the second list.
12. A data storage system as claimed in claim 9, wherein a flag is provided with each data reference in the first list indicating whether or not the data has been read whilst on the first list.

13. A data storage system as claimed in claim 9, wherein, if the data was read when referenced in the first list, the data is added to the head of the second list when the data is destaged.
14. A data storage system as claimed in claim 9, wherein, if the data was not read when referenced in the first list, the data is either maintained in its position in the second list or discarded.
15. A data storage system as claimed in claim 12, wherein the flag includes a timestamp each time the data is read and the timestamp is used to prioritise the position of the data reference in the second list.
16. A data storage system as claimed in claim 9, wherein data is partly dirty and partly clean and is referenced in both the first and second lists.
17. A computer program product stored on a computer readable storage medium, comprising computer readable program code means for retaining data in a data cache by performing the steps of:
 - referencing dirty data stored in a cache in a first least recently used list;
 - and
 - referencing clean data in the cache in a second least recently used list;
 - wherein dirty data is destaged from the cache when it reaches the tail of the first least recently used list and clean data is purged from the cache when it reaches the tail of the second least recently used list.